# House Pricing Report

Viacheslav Simonov 1/20/2022

# House Pricing Prediction

## Introduction

The main goal of this project is to predict correct prices for house.

This Data Science competition is offered by kaggle.com. Detail info can be found **here**.

## Data

The Ames Housing dataset was compiled by Dean De Cock for use in data science education. It's an incredible alternative for data scientists looking for a modernized and expanded version of the often cited Boston Housing dataset. The dataset contains 79 explanatory variables describing (almost) every aspect of residential homes in Ames, Iowa, this competition challenges you to predict the final price of each home.

Detailde explanation of all data columns is provided below:

```
cat(readLines('data/data_description.txt'), sep = '\n')
```

```
MSSubClass: Identifies the type of dwelling involved in the sale.

        20  1-STORY 1946 & NEWER ALL STYLES
        30  1-STORY 1945 & OLDER
        40  1-STORY W/FINISHED ATTIC ALL AGES
        45  1-1/2 STORY - UNFINISHED ALL AGES
        50  1-1/2 STORY FINISHED ALL AGES
        60  2-STORY 1946 & NEWER
        70  2-STORY 1945 & OLDER
        75  2-1/2 STORY ALL AGES
        80  SPLIT OR MULTI-LEVEL
        85  SPLIT FOYER
        90  DUPLEX - ALL STYLES AND AGES
       120  1-STORY PUD (Planned Unit Development) - 1946 & NEWER
       150  1-1/2 STORY PUD - ALL AGES
       160  2-STORY PUD - 1946 & NEWER
       180  PUD - MULTILEVEL - INCL SPLIT LEV/FOYER
       190  2 FAMILY CONVERSION - ALL STYLES AND AGES

MSZoning: Identifies the general zoning classification of the sale.

       A    Agriculture
       C    Commercial
       FV   Floating Village Residential
       I    Industrial
       RH   Residential High Density
       RL   Residential Low Density
       RP   Residential Low Density Park
       RM   Residential Medium Density

LotFrontage: Linear feet of street connected to property

LotArea: Lot size in square feet

Street: Type of road access to property

       Grvl Gravel
       Pave Paved

Alley: Type of alley access to property

       Grvl Gravel
```

```
       Pave Paved
       NA   No alley access

LotShape: General shape of property

       Reg  Regular
       IR1  Slightly irregular
       IR2  Moderately Irregular
       IR3  Irregular

LandContour: Flatness of the property

       Lvl  Near Flat/Level
       Bnk  Banked - Quick and significant rise from street grade to building
       HLS  Hillside - Significant slope from side to side
       Low  Depression

Utilities: Type of utilities available

       AllPub   All public Utilities (E,G,W,& S)
       NoSewr   Electricity, Gas, and Water (Septic Tank)
       NoSeWa   Electricity and Gas Only
       ELO  Electricity only

LotConfig: Lot configuration

       Inside   Inside lot
       Corner   Corner lot
       CulDSac  Cul-de-sac
       FR2  Frontage on 2 sides of property
       FR3  Frontage on 3 sides of property

LandSlope: Slope of property

       Gtl  Gentle slope
       Mod  Moderate Slope
       Sev  Severe Slope

Neighborhood: Physical locations within Ames city limits

       Blmngtn  Bloomington Heights
       Blueste  Bluestem
       BrDale   Briardale
       BrkSide  Brookside
       ClearCr  Clear Creek
       CollgCr  College Creek
       Crawfor  Crawford
       Edwards  Edwards
       Gilbert  Gilbert
       IDOTRR   Iowa DOT and Rail Road
       MeadowV  Meadow Village
       Mitchel  Mitchell
       Names    North Ames
       NoRidge  Northridge
       NPkVill  Northpark Villa
       NridgHt  Northridge Heights
       NWAmes   Northwest Ames
       OldTown  Old Town
       SWISU    South & West of Iowa State University
       Sawyer   Sawyer
       SawyerW  Sawyer West
       Somerst  Somerset
       StoneBr  Stone Brook
       Timber   Timberland
       Veenker  Veenker
```

```
Condition1: Proximity to various conditions

       Artery   Adjacent to arterial street
       Feedr    Adjacent to feeder street
       Norm Normal
       RRNn Within 200' of North-South Railroad
       RRAn Adjacent to North-South Railroad
       PosN Near positive off-site feature--park, greenbelt, etc.
       PosA Adjacent to postive off-site feature
       RRNe Within 200' of East-West Railroad
       RRAe Adjacent to East-West Railroad

Condition2: Proximity to various conditions (if more than one is present)

       Artery   Adjacent to arterial street
       Feedr    Adjacent to feeder street
       Norm Normal
       RRNn Within 200' of North-South Railroad
       RRAn Adjacent to North-South Railroad
       PosN Near positive off-site feature--park, greenbelt, etc.
       PosA Adjacent to postive off-site feature
       RRNe Within 200' of East-West Railroad
       RRAe Adjacent to East-West Railroad

BldgType: Type of dwelling

       1Fam Single-family Detached
       2FmCon   Two-family Conversion; originally built as one-family dwelling
       Duplx    Duplex
       TwnhsE   Townhouse End Unit
       TwnhsI   Townhouse Inside Unit

HouseStyle: Style of dwelling

       1Story   One story
       1.5Fin   One and one-half story: 2nd level finished
       1.5Unf   One and one-half story: 2nd level unfinished
       2Story   Two story
       2.5Fin   Two and one-half story: 2nd level finished
       2.5Unf   Two and one-half story: 2nd level unfinished
       SFoyer   Split Foyer
       SLvl Split Level

OverallQual: Rates the overall material and finish of the house

       10   Very Excellent
       9    Excellent
       8    Very Good
       7    Good
       6    Above Average
       5    Average
       4    Below Average
       3    Fair
       2    Poor
       1    Very Poor

OverallCond: Rates the overall condition of the house

       10   Very Excellent
       9    Excellent
       8    Very Good
       7    Good
       6    Above Average
       5    Average
       4    Below Average
```

```
       3	Fair
       2	Poor
       1	Very Poor
		
YearBuilt: Original construction date

YearRemodAdd: Remodel date (same as construction date if no remodeling or additions)

RoofStyle: Type of roof

       Flat	Flat
       Gable	Gable
       Gambrel	Gabrel (Barn)
       Hip	Hip
       Mansard	Mansard
       Shed	Shed
		
RoofMatl: Roof material

       ClyTile	Clay or Tile
       CompShg	Standard (Composite) Shingle
       Membran	Membrane
       Metal	Metal
       Roll	Roll
       Tar&Grv	Gravel & Tar
       WdShake	Wood Shakes
       WdShngl	Wood Shingles
		
Exterior1st: Exterior covering on house

       AsbShng	Asbestos Shingles
       AsphShn	Asphalt Shingles
       BrkComm	Brick Common
       BrkFace	Brick Face
       CBlock	Cinder Block
       CemntBd	Cement Board
       HdBoard	Hard Board
       ImStucc	Imitation Stucco
       MetalSd	Metal Siding
       Other	Other
       Plywood	Plywood
       PreCast	PreCast
       Stone	Stone
       Stucco	Stucco
       VinylSd	Vinyl Siding
       Wd Sdng	Wood Siding
       WdShing	Wood Shingles
	
Exterior2nd: Exterior covering on house (if more than one material)

       AsbShng	Asbestos Shingles
       AsphShn	Asphalt Shingles
       BrkComm	Brick Common
       BrkFace	Brick Face
       CBlock	Cinder Block
       CemntBd	Cement Board
       HdBoard	Hard Board
       ImStucc	Imitation Stucco
       MetalSd	Metal Siding
       Other	Other
       Plywood	Plywood
       PreCast	PreCast
       Stone	Stone
       Stucco	Stucco
       VinylSd	Vinyl Siding
       Wd Sdng	Wood Siding
```

```
       Wd Sdng  Wood Siding
       WdShing  Wood Shingles

MasVnrType: Masonry veneer type

       BrkCmn   Brick Common
       BrkFace  Brick Face
       CBlock   Cinder Block
       None None
       Stone    Stone

MasVnrArea: Masonry veneer area in square feet

ExterQual: Evaluates the quality of the material on the exterior

       Ex   Excellent
       Gd   Good
       TA   Average/Typical
       Fa   Fair
       Po   Poor

ExterCond: Evaluates the present condition of the material on the exterior

       Ex   Excellent
       Gd   Good
       TA   Average/Typical
       Fa   Fair
       Po   Poor

Foundation: Type of foundation

       BrkTil   Brick & Tile
       CBlock   Cinder Block
       PConc    Poured Contrete
       Slab Slab
       Stone    Stone
       Wood Wood

BsmtQual: Evaluates the height of the basement

       Ex   Excellent (100+ inches)
       Gd   Good (90-99 inches)
       TA   Typical (80-89 inches)
       Fa   Fair (70-79 inches)
       Po   Poor (<70 inches
       NA   No Basement

BsmtCond: Evaluates the general condition of the basement

       Ex   Excellent
       Gd   Good
       TA   Typical - slight dampness allowed
       Fa   Fair - dampness or some cracking or settling
       Po   Poor - Severe cracking, settling, or wetness
       NA   No Basement

BsmtExposure: Refers to walkout or garden level walls

       Gd   Good Exposure
       Av   Average Exposure (split levels or foyers typically score average or above)
       Mn   Mimimum Exposure
       No   No Exposure
       NA   No Basement

BsmtFinType1: Rating of basement finished area
```

```
       GLQ  Good Living Quarters
       ALQ  Average Living Quarters
       BLQ  Below Average Living Quarters
       Rec  Average Rec Room
       LwQ  Low Quality
       Unf  Unfinshed
       NA   No Basement

BsmtFinSF1: Type 1 finished square feet

BsmtFinType2: Rating of basement finished area (if multiple types)

       GLQ  Good Living Quarters
       ALQ  Average Living Quarters
       BLQ  Below Average Living Quarters
       Rec  Average Rec Room
       LwQ  Low Quality
       Unf  Unfinshed
       NA   No Basement

BsmtFinSF2: Type 2 finished square feet

BsmtUnfSF: Unfinished square feet of basement area

TotalBsmtSF: Total square feet of basement area

Heating: Type of heating

       Floor    Floor Furnace
       GasA Gas forced warm air furnace
       GasW Gas hot water or steam heat
       Grav Gravity furnace
       OthW Hot water or steam heat other than gas
       Wall Wall furnace

HeatingQC: Heating quality and condition

       Ex   Excellent
       Gd   Good
       TA   Average/Typical
       Fa   Fair
       Po   Poor

CentralAir: Central air conditioning

       N    No
       Y    Yes

Electrical: Electrical system

       SBrkr    Standard Circuit Breakers & Romex
       FuseA    Fuse Box over 60 AMP and all Romex wiring (Average)
       FuseF    60 AMP Fuse Box and mostly Romex wiring (Fair)
       FuseP    60 AMP Fuse Box and mostly knob & tube wiring (poor)
       Mix  Mixed

1stFlrSF: First Floor square feet

2ndFlrSF: Second floor square feet

LowQualFinSF: Low quality finished square feet (all floors)

GrLivArea: Above grade (ground) living area square feet

BsmtFullBath: Basement full bathrooms
```

```
BsmtHalfBath: Basement half bathrooms

FullBath: Full bathrooms above grade

HalfBath: Half baths above grade

Bedroom: Bedrooms above grade (does NOT include basement bedrooms)

Kitchen: Kitchens above grade

KitchenQual: Kitchen quality

       Ex   Excellent
       Gd   Good
       TA   Typical/Average
       Fa   Fair
       Po   Poor

TotRmsAbvGrd: Total rooms above grade (does not include bathrooms)

Functional: Home functionality (Assume typical unless deductions are warranted)

       Typ  Typical Functionality
       Min1 Minor Deductions 1
       Min2 Minor Deductions 2
       Mod  Moderate Deductions
       Maj1 Major Deductions 1
       Maj2 Major Deductions 2
       Sev  Severely Damaged
       Sal  Salvage only

Fireplaces: Number of fireplaces

FireplaceQu: Fireplace quality

       Ex   Excellent - Exceptional Masonry Fireplace
       Gd   Good - Masonry Fireplace in main level
       TA   Average - Prefabricated Fireplace in main living area or Masonry Fireplace in basement
       Fa   Fair - Prefabricated Fireplace in basement
       Po   Poor - Ben Franklin Stove
       NA   No Fireplace

GarageType: Garage location

       2Types   More than one type of garage
       Attchd   Attached to home
       Basment  Basement Garage
       BuiltIn  Built-In (Garage part of house - typically has room above garage)
       CarPort  Car Port
       Detchd   Detached from home
       NA   No Garage

GarageYrBlt: Year garage was built

GarageFinish: Interior finish of the garage

       Fin  Finished
       RFn  Rough Finished
       Unf  Unfinished
       NA   No Garage

GarageCars: Size of garage in car capacity

GarageArea: Size of garage in square feet
```

```
GarageQual: Garage quality

       Ex   Excellent
       Gd   Good
       TA   Typical/Average
       Fa   Fair
       Po   Poor
       NA   No Garage

GarageCond: Garage condition

       Ex   Excellent
       Gd   Good
       TA   Typical/Average
       Fa   Fair
       Po   Poor
       NA   No Garage

PavedDrive: Paved driveway

       Y    Paved
       P    Partial Pavement
       N    Dirt/Gravel

WoodDeckSF: Wood deck area in square feet

OpenPorchSF: Open porch area in square feet

EnclosedPorch: Enclosed porch area in square feet

3SsnPorch: Three season porch area in square feet

ScreenPorch: Screen porch area in square feet

PoolArea: Pool area in square feet

PoolQC: Pool quality

       Ex   Excellent
       Gd   Good
       TA   Average/Typical
       Fa   Fair
       NA   No Pool

Fence: Fence quality

       GdPrv    Good Privacy
       MnPrv    Minimum Privacy
       GdWo Good Wood
       MnWw Minimum Wood/Wire
       NA   No Fence

MiscFeature: Miscellaneous feature not covered in other categories

       Elev Elevator
       Gar2 2nd Garage (if not described in garage section)
       Othr Other
       Shed Shed (over 100 SF)
       TenC Tennis Court
       NA   None

MiscVal: $Value of miscellaneous feature

MoSold: Month Sold (MM)
```

```
YrSold: Year Sold (YYYY)

SaleType: Type of sale

       WD   Warranty Deed - Conventional
       CWD  Warranty Deed - Cash
       VWD  Warranty Deed - VA Loan
       New  Home just constructed and sold
       COD  Court Officer Deed/Estate
       Con  Contract 15% Down payment regular terms
       ConLw    Contract Low Down payment and low interest
       ConLI    Contract Low Interest
       ConLD    Contract Low Down
       Oth  Other

SaleCondition: Condition of sale

       Normal   Normal Sale
       Abnorml  Abnormal Sale -  trade, foreclosure, short sale
       AdjLand  Adjoining Land Purchase
       Alloca   Allocation - two linked properties with separate deeds, typically condo with a garage unit
       Family   Sale between family members
       Partial  Home was not completed when last assessed (associated with New Homes)
```

## Install dependencies and parse data

```
if(!require(caret)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: caret

## Loading required package: lattice

## Loading required package: ggplot2
```

```
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: tidyverse

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✓ tibble  3.1.4     ✓ dplyr   1.0.7
## ✓ tidyr   1.1.3     ✓ stringr 1.4.0
## ✓ readr   2.0.1     ✓ forcats 0.5.1
## ✓ purrr   0.3.4

## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## x purrr::lift()   masks caret::lift()
```

```
if(!require(ggplot2)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
if(!require(infotheo)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: infotheo
```

```
if(!require(mboost)) install.packages("mboost", repos = "http://cran.us.r-project.org")
```

```
## Loading required package: mboost

## Loading required package: parallel

## Loading required package: stabs

##
## Attaching package: 'mboost'

## The following object is masked from 'package:tidyr':
##
##     extract

## The following object is masked from 'package:ggplot2':
##
##     %+%
```

```
library(caret)
library(tidyverse)
library(ggplot2)
library(infotheo)
library(mboost)
```

```
train_set <- read.csv("data/train.csv", stringsAsFactors = T)
goal_set <- read.csv("data/test.csv", stringsAsFactors = T)

whole_set <- bind_rows(train_set, goal_set)
```

## Data wrangling

In order to get better results for ML models, we need to create some new variables, first of all, I would like to summarise some divided variables to one common, such as Porch Area, Basement Area etc. Also, I would like to extract some really significant variables as Overall Quality and Condition to even more important using them together, also it seems reasonable to change variables related to years and to change Build year to Age for instance. For this purpose we will use the next function.

As score on kaggle.com is estimated with log of SalePrice, I will transform SalePrices in this way also.

```
wrangleData <- function(dataset) {
  qualityRateColumns <- c("ExterCond", "ExterQual", "BsmtCond", "BsmtQual", "HeatingQC", "KitchenQual", "FireplaceQu", "GarageQual",
  informativeNAColumns <- c("Alley", "MasVnrType", "BsmtExposure", "GarageType", "MiscFeature", "BsmtFinType1", "BsmtFinType2", "Ele
  meanIfNAColumns <- c("LotFrontage")
  zeroIfNAColumns <- c("BsmtFinSF1", "BsmtFinSF2", "BsmtUnfSF", "TotalBsmtSF", "BsmtFullBath", "BsmtHalfBath", "GarageCars", "Garage

  # Set numeric rating to factor columns
  for (col in qualityRateColumns) {
    dataset[[col]] <- condQualityToInt(dataset[[col]])
  }

  # Add NA factor
  for (col in informativeNAColumns) {
    dataset[[col]] <- addNA(dataset[[col]])
  }

  # Set mean instead of NA to the columns that require it
  for (col in meanIfNAColumns) {
    dataset[[col]][which(is.na(dataset[[col]]))] <- mean(dataset[[col]], na.rm = T)
  }

  # Set zero instead of NA to the columns that require it
  for (col in zeroIfNAColumns) {
    dataset[[col]][which(is.na(dataset[[col]]))] <- 0
  }

  # Convert 2 level factor to numeric col as obviously Y is good and N level is bad
  dataset$CentralAir <- sapply(dataset$CentralAir, yesNoToBinary)
```

```
  # Set other factor to SaleType if NA
  dataset$SaleType[which(is.na(dataset$SaleType))] <- factor("Oth")


  # Define overall number of Bathrooms
  dataset$Bathrooms <- dataset$BsmtFullBath+dataset$BsmtHalfBath*0.5+dataset$FullBath+dataset$HalfBath*0.5

  dataset$BsmtFinSF <- dataset$BsmtFinSF1 + dataset$BsmtFinSF2

  dataset$TotalSquare <- dataset$TotalBsmtSF + dataset$X1stFlrSF + dataset$X2ndFlrSF

  # Compute age
  dataset$Age <- dataset$YrSold - dataset$YearBuilt
  # Compute age of renovation
  dataset$SinceRenov <- ifelse(dataset$YrSold - dataset$YearRemodAdd < 0, 0, dataset$YrSold - dataset$YearRemodAdd)
  dataset$GarageAge <- dataset$YrSold - dataset$GarageYrBlt

  dataset$Freshness <- dataset$Age * dataset$SinceRenov
  dataset$Newness <- sqrt(dataset$SinceRenov * dataset$GrLivArea)

  dataset$New <- ifelse(dataset$Age == 0, 1, 0)
  dataset$Fresh <- ifelse(dataset$SinceRenov == 0, 1, 0)

  dataset$Overall <- dataset$OverallCond * dataset$OverallQual
  dataset$ExternalOverall <- dataset$ExterCond * dataset$ExterQual
  dataset$GarageOverall <- dataset$GarageQual * dataset$GarageCond

  dataset$LotArea_log <- log(dataset$LotArea)

  dataset$Spaciousness <- (dataset$X1stFlrSF + dataset$X2ndFlrSF)/dataset$TotRmsAbvGrd

  # COmpute overall porch area
  dataset$PorchArea <- dataset$WoodDeckSF + dataset$OpenPorchSF+ dataset$EnclosedPorch+ dataset$X3SsnPorch+ dataset$ScreenPorch

  # Compute WOW effect for basement, garage and house
  dataset$GarageWow <- dataset$GarageArea * dataset$GarageQual * dataset$GarageCond
  dataset$OverallWow <- dataset$OverallQual * dataset$OverallCond * dataset$GrLivArea
  dataset$BasementWow <- dataset$BsmtQual * dataset$BsmtCond * dataset$BsmtFinSF

  dataset$SalePrice_Log <- ifelse(is.na(dataset$SalePrice), 0, log(dataset$SalePrice))

  dataset %>% select(-WoodDeckSF, -OpenPorchSF, -EnclosedPorch, -X3SsnPorch, -ScreenPorch, -X1stFlrSF, -X2ndFlrSF, -YearBuilt, -YrSo
}

convertFactorsToBinaryColumns <- function(dataset, factor_columns = colnames(dataset)) {
  for (col in factor_columns) {
    column <- dataset[[col]]
    if (class(column) == "factor") {
      for (level in levels(column)) {
        if (!is.na(level)) {
          binaryColumn <- paste(col, str_remove_all(level, " "), sep = "_")
          dataset[[binaryColumn]] <- as.numeric(column == level)
        }
      }
      dataset <- dataset %>% select(-col)
    }
  }

  dataset
}

addNaFactor <- function(vector) {
  vector <- as.character(vector)
  vector[which(is.na(vector))] <- "NA"
```

```
  as.factor(vector)
}

yearToFactor <- function(yearVec) {
  as.factor(sapply(yearVec, function(year) {
    if (is.na(year)) {
      result <- "NA"
    } else if (year > 2000) {
      result <- "After 2000"
    } else if (year > 1980) {
      result <- "1981-2000"
    } else if (year > 1960) {
      result <- "1961-1980"
    } else if (year > 1940) {
      result <- "1941-1960"
    } else {
      result <- "Before 1940"
    }

    result
  }))
}

yesNoToBinary <- function(fact) {
  ifelse(fact == "Y", 1, 0)
}

condQualityToInt <- function(fact) {
  charVec <- as.character(fact)

  sapply(charVec, function(qual) {
    if (is.na(qual)) {
      result <- 0
    } else if (qual == "Ex") {
      result <- 5
    } else if (qual == "Gd") {
      result <- 4
    } else if (qual == "TA") {
      result <- 3
    } else if (qual == "Fa") {
      result <- 2
    } else if (qual == "Po") {
      result <- 1
    } else {
      result <- 0
    }

    result
  })
}

doubleInfoColumnsToDummies <- function(dataset, double_columns, new_column_prefix) {
  column_1 <- dataset[[double_columns[1]]]
  column_2 <- dataset[[double_columns[2]]]

  all_levels <- unique(c(levels(column_1), levels(column_2)))
  for (level in all_levels) {
    if (!is.na(level)) {
      binaryColumn <- paste(new_column_prefix, str_remove_all(level, " "), sep = "_")
      dataset[[binaryColumn]] <- as.numeric(column_1 == level | column_2 == level)
    }
  }

  dataset
}
```

```
}

RMSE <- function(true_ratings, predicted_ratings){
  sqrt(mean((true_ratings - predicted_ratings)^2))
}
```

```
engineered_whole_set <- wrangleData(whole_set)
```

The next thing is to check how our freshly created variables correlated with our goal value of SalePrice. For this purpose I will plot all new variables against target variable.

```
new_numeric_vars <- c("TotalSquare","Bathrooms","Age","SinceRenov","GarageAge","Freshness","Newness", "Overall","ExternalOverall","C

new_level_vars <- c("New", "Fresh")

engineered_train_set <- engineered_whole_set %>% filter(SalePrice_Log > 0)

for (var in new_level_vars) {
  print(engineered_train_set %>%
          ggplot(aes(x = .data[[var]], y = SalePrice_Log, group= .data[[var]])) + geom_boxplot())
}
```

```
for (var in new_numeric_vars) {
  print(engineered_train_set %>%
          ggplot(aes(x = .data[[var]], y = SalePrice_Log)) + geom_point())
}
```

## Mutual information analysis and drop of not important predictors

In order to have an opportunity to extract not important features of the base once, we should use mutual information analysis before new feature extracting that will lead to multiplying of the predictors amount and it won't be easy to define redudant predictors.

```
mi_scores <- data.frame(col_name = colnames(engineered_train_set), mi = sapply(colnames(engineered_train_set), function(col_name) {
  mutinformation(X = as.integer(engineered_train_set[[col_name]]), Y = engineered_train_set$SalePrice)
}))

mi_scores %>% filter(!(col_name %in% c("SalePrice_Log", "SalePrice", "Id"))) %>% arrange(desc(mi)) %>% tail(30)
```

```
##                         col_name          mi
## BsmtFinType2   BsmtFinType2 0.399378077
## Condition1       Condition1 0.380689100
## BldgType           BldgType 0.379890083
## Fence                 Fence 0.375979445
## RoofStyle         RoofStyle 0.358190163
## GarageOverall GarageOverall 0.352836534
## BsmtCond           BsmtCond 0.298650563
## GarageQual       GarageQual 0.288711970
## LandContour     LandContour 0.277516693
## GarageCond       GarageCond 0.260239269
## Fresh                 Fresh 0.242511473
## ExterCond         ExterCond 0.237317602
## Electrical       Electrical 0.220084524
## Functional       Functional 0.217802241
## PavedDrive       PavedDrive 0.197436550
## MiscVal             MiscVal 0.179635557
## Alley                 Alley 0.166222234
## New                     New 0.159889436
## CentralAir       CentralAir 0.157974688
## LandSlope         LandSlope 0.149264155
## KitchenAbvGr   KitchenAbvGr 0.119027610
## LowQualFinSF   LowQualFinSF 0.113325901
## MiscFeature     MiscFeature 0.107683402
## Heating             Heating 0.095033803
## RoofMatl           RoofMatl 0.082170085
## Condition2       Condition2 0.055250025
## PoolArea           PoolArea 0.029648425
## PoolQC               PoolQC 0.025491969
## Street               Street 0.021417742
## Utilities         Utilities 0.003823618
```

```
engineered_whole_set <- engineered_whole_set %>% select(-SalePrice, -BldgType, -Fence, -RoofStyle, -BsmtCond, -LandContour, -PoolQC,
```

## Double columns to dummies

Next step in the data wrangling is to summarise columns that divided for 2 different columns, for Condition and Exterior, I would like to just turn them into binary vectors for each factor level, but for BasementFinType, instead of 1s, I would like to store square feet of the territory, so for the first two, we will use the helper function, and for the third we will write a separate script.

```
doubleInfoColumnsToDummies <- function(dataset, double_columns, new_column_prefix) {
  column_1 <- dataset[[double_columns[1]]]
  column_2 <- dataset[[double_columns[2]]]

  all_levels <- unique(c(levels(column_1), levels(column_2)))
  for (level in all_levels) {
    if (!is.na(level)) {
      binaryColumn <- paste(new_column_prefix, str_remove_all(level, " "), sep = "_")
      dataset[[binaryColumn]] <- as.numeric(column_1 == level | column_2 == level)
    }
  }

  dataset
}

engineered_whole_set <- doubleInfoColumnsToDummies(engineered_whole_set, c("Condition1", "Condition2"), "Condition")
engineered_whole_set <- doubleInfoColumnsToDummies(engineered_whole_set, c("Exterior1st", "Exterior2nd"), "Ext")

bsmt_type_1 <- engineered_whole_set[["BsmtFinType1"]]
bsmt_type_2 <- engineered_whole_set[["BsmtFinType2"]]

all_levels <- unique(c(levels(bsmt_type_1), levels(bsmt_type_2)))

for (level in all_levels) {
  if (!is.na(level)) {
    bsmt1Vector <- as.numeric(bsmt_type_1 == level) * engineered_whole_set$BsmtFinSF1
    bsmt2Vector <- as.numeric(bsmt_type_2 == level) * engineered_whole_set$BsmtFinSF2

    summaryColumn <- paste("BF", str_remove_all(level, " "), sep = "_")
    engineered_whole_set[[summaryColumn]] <- bsmt1Vector + bsmt2Vector
  }
}

rm(bsmt1Vector, bsmt2Vector, summaryColumn, all_levels, level)
```

After that, we can drop old columns from which we took the data.

```
engineered_whole_set <- engineered_whole_set %>% select(-"Condition1", -"Condition2", -"Exterior1st", -"Exterior2nd", -"BsmtFinSF1",
```

## Enginering clustering features

The next data-engineering step is to run K-means algorithm in order to define cluster withing the data using the most important predictors. I will define 10 clusters, and new features will be euclidian disctance to the center of the particular cluster.

```
set_for_clustering <- engineered_whole_set %>% select(OverallWow, LotArea, TotalSquare, GrLivArea, Spaciousness, Age, SinceRenov, Po
k_m <- kmeans(set_for_clustering, centers = 10, iter.max = 30)

for (row in 1:nrow(k_m[["centers"]])) {
  columnName <- paste("Centroid", row, sep = "_")
  engineered_whole_set[[columnName]] <- sqrt(rowSums(sweep(as.matrix(set_for_clustering), 2, k_m[["centers"]][row,])**2))
}
```

## Convert factor columns to dummies(binary columns)

In order to have all the predictors as numeric columns, we need to convert factor columns to binary numeric columns, I will implement it with the helper function

```
convertFactorsToBinaryColumns <- function(dataset, factor_columns = colnames(dataset)) {
  for (col in factor_columns) {
    column <- dataset[[col]]
    if (class(column) == "factor") {
      for (level in levels(column)) {
        if (!is.na(level)) {
          binaryColumn <- paste(col, str_remove_all(level, " "), sep = "_")
          dataset[[binaryColumn]] <- as.numeric(column == level)
        }
      }
      dataset <- dataset %>% select(-col)
    }
  }

  dataset
}

engineered_whole_set <- convertFactorsToBinaryColumns(engineered_whole_set)
```

```
## Note: Using an external vector in selections is ambiguous.
## ℹ Use `all_of(col)` instead of `col` to silence this message.
## ℹ See <https://tidyselect.r-lib.org/reference/faq-external-vector.html>.
## This message is displayed once per session.
```

## Scaling data

```
Ids <- engineered_whole_set$Id
SalePrices <- engineered_whole_set$SalePrice_Log

engineered_whole_set <- engineered_whole_set %>%
  select(-Id, -SalePrice_Log)

engineered_whole_set <- as.data.frame(scale(engineered_whole_set))

engineered_whole_set$Id <- Ids
engineered_whole_set$SalePrice_Log <- SalePrices

rm(Ids, SalePrices)
```

Now we have 209 predictors, and our data looks like this:

```
engineered_train_set <- engineered_whole_set %>% filter(SalePrice_Log > 0)

head(engineered_train_set)
```

```
##    MSSubClass LotFrontage     LotArea OverallQual OverallCond MasVnrArea
## 1  0.06731988 -0.20203292 -0.21784137  0.64607270  -0.5071973  0.5289435
## 2 -0.87346638  0.50178450 -0.07203174 -0.06317371   2.1879039 -0.5669188
## 3  0.06731988 -0.06126943  0.13717338  0.64607270  -0.5071973  0.3388450
## 4  0.30251644 -0.43663872 -0.07837129  0.64607270  -0.5071973 -0.5669188
## 5  0.06731988  0.68946915  0.51881423  1.35531911  -0.5071973  1.3899782
## 6 -0.16787668  0.73639031  0.50042953 -0.77242013  -0.5071973 -0.5669188
##    ExterQual  ExterCond   BsmtQual   BsmtUnfSF TotalBsmtSF  HeatingQC
## 1  1.0396273 -0.2300074  0.5769954 -0.93400478  -0.4430020  0.8854676
## 2 -0.6836391 -0.2300074  0.5769954 -0.62917590   0.4773814  0.8854676
## 3  1.0396273 -0.2300074  0.5769954 -0.28794954  -0.2979169  0.8854676
## 4 -0.6836391 -0.2300074 -0.5274306 -0.04681624  -0.6696974 -0.1584257
## 5  1.0396273 -0.2300074  0.5769954 -0.16055836   0.2121478  0.8854676
## 6 -0.6836391 -0.2300074  0.5769954 -1.12964123  -0.5790193  0.8854676
##   CentralAir LowQualFinSF  GrLivArea BedroomAbvGr KitchenAbvGr KitchenQual
## 1  0.2682439   -0.1011797  0.4134764     0.169898   -0.2076629   0.7368952
## 2  0.2682439   -0.1011797 -0.4718098     0.169898   -0.2076629  -0.7662474
```

```
## 3  0.2682439   -0.1011797  0.5636589     0.169898   -0.2076629   0.7368952
## 4  0.2682439   -0.1011797  0.4273090     0.169898   -0.2076629   0.7368952
## 5  0.2682439   -0.1011797  1.3778060     1.385418   -0.2076629   0.7368952
## 6  0.2682439   -0.1011797 -0.2742013    -2.261142   -0.2076629  -0.7662474
##   TotRmsAbvGrd Fireplaces FireplaceQu GarageYrBlt GarageCars  GarageArea
## 1    0.9866803 -0.9241529  -0.9786628   0.2949519  0.3069872  0.34930377
## 2   -0.2877090  0.6235248   0.6818972   0.2349100  0.3069872 -0.05898129
## 3   -0.2877090  0.6235248   0.6818972   0.2905044  0.3069872  0.62767994
## 4    0.3494857  0.6235248   1.2354172   0.2838330  1.6189865  0.78542644
## 5    1.6238750  0.6235248   0.6818972   0.2882806  1.6189865  1.68550941
## 6   -0.9249036 -0.9241529  -0.9786628   0.2727142  0.3069872  0.03381077
##   GarageQual GarageCond     MiscVal     MoSold  Bathrooms   BsmtFinSF
## 1  0.2780431  0.2682975 -0.08957661 -1.5519176  1.5844947  0.45087657
## 2  0.2780431  0.2682975 -0.08957661 -0.4468483  0.3481568  1.02085646
## 3  0.2780431  0.2682975 -0.08957661  1.0265775  1.5844947 -0.01013657
## 4  0.2780431  0.2682975 -0.08957661 -1.5519176 -0.2700121 -0.57592543
## 5  0.2780431  0.2682975 -0.08957661  2.1316468  1.5844947  0.34400534
## 6  0.2780431  0.2682975  1.14411615  1.3949339  0.3481568  0.50535994
##   TotalSquare        Age SinceRenov  GarageAge  Freshness    Newness        New
## 1  0.02299941 -1.0377034 -0.8869899 -0.2944807 -0.7931369 -0.6801601 -0.2033963
## 2 -0.02916668 -0.1806410  0.3575969 -0.2366646 -0.1856179  0.4824205 -0.2033963
## 3  0.19688635 -0.9717755 -0.8391212 -0.2900333 -0.7821029 -0.5581707 -0.2033963
## 4 -0.09251121  1.7971952  0.5969405 -0.2878096  1.3169531  1.0434460 -0.2033963
## 5  0.98807193 -0.9388116 -0.7433837 -0.2878096 -0.7678236 -0.2371173 -0.2033963
## 6 -0.48375682 -0.6751001 -0.4561714 -0.2700200 -0.6639742 -0.1766049 -0.2033963
##        Fresh    Overall ExternalOverall GarageOverall LotArea_log Spaciousness
## 1 -0.2999362  0.1376120       0.6957741      0.281194  -0.1036605   -0.3671666
## 2 -0.2999362  1.5527776      -0.6842698      0.281194   0.1465458   -0.4409755
## 3 -0.2999362  0.1376120       0.6957741      0.281194   0.4575570    1.4456518
## 4 -0.2999362  0.1376120      -0.6842698      0.281194   0.1363060    0.3140869
## 5 -0.2999362  0.6819064       0.6957741      0.281194   0.9224713    0.2911127
## 6 -0.2999362 -0.9509770      -0.6842698      0.281194   0.9024300    0.8998260
##    PorchArea   GarageWow OverallWow BasementWow Condition_Artery
## 1 -0.7621454  0.34461591  0.2639134  0.46064229       -0.1833813
## 2  0.7189065 -0.03732531  0.2897886  0.98206667       -0.1833813
## 3 -0.8808795  0.60503039  0.3587180  0.03890197       -0.1833813
## 4  0.7751490  0.75259859  0.2726454 -0.47868841       -0.1833813
## 5  0.5814249  1.59460539  1.2643522  0.36287521       -0.1833813
## 6  1.2938296  0.04947951 -0.6556207  0.51048432       -0.1833813
##   Condition_Feedr Condition_Norm Condition_PosA Condition_PosN Condition_RRAe
## 1      -0.2509565      0.1018855    -0.08511107     -0.1163487     -0.1001557
## 2       3.9833898      0.1018855    -0.08511107     -0.1163487     -0.1001557
## 3      -0.2509565      0.1018855    -0.08511107     -0.1163487     -0.1001557
## 4      -0.2509565      0.1018855    -0.08511107     -0.1163487     -0.1001557
## 5      -0.2509565      0.1018855    -0.08511107     -0.1163487     -0.1001557
## 6      -0.2509565      0.1018855    -0.08511107     -0.1163487     -0.1001557
##   Condition_RRAn Condition_RRNe Condition_RRNn Ext_AsbShng Ext_AsphShn
## 1     -0.1333279     -0.0453765    -0.06149287  -0.1279035 -0.03703704
## 2     -0.1333279     -0.0453765    -0.06149287  -0.1279035 -0.03703704
## 3     -0.1333279     -0.0453765    -0.06149287  -0.1279035 -0.03703704
## 4     -0.1333279     -0.0453765    -0.06149287  -0.1279035 -0.03703704
## 5     -0.1333279     -0.0453765    -0.06149287  -0.1279035 -0.03703704
## 6     -0.1333279     -0.0453765    -0.06149287  -0.1279035 -0.03703704
##   Ext_BrkComm Ext_BrkFace  Ext_CBlock Ext_CemntBd Ext_HdBoard Ext_ImStucc
## 1  -0.0453765  -0.1783325 -0.03703704  -0.2123613   -0.435226 -0.07185764
## 2  -0.0453765  -0.1783325 -0.03703704  -0.2123613   -0.435226 -0.07185764
## 3  -0.0453765  -0.1783325 -0.03703704  -0.2123613   -0.435226 -0.07185764
## 4  -0.0453765  -0.1783325 -0.03703704  -0.2123613   -0.435226 -0.07185764
## 5  -0.0453765  -0.1783325 -0.03703704  -0.2123613   -0.435226 -0.07185764
## 6  -0.0453765  -0.1783325 -0.03703704  -0.2123613   -0.435226 -0.07185764
##   Ext_MetalSd Ext_Plywood   Ext_Stone Ext_Stucco Ext_VinylSd Ext_WdSdng
## 1  -0.4324394  -0.3415252 -0.04902063 -0.1411004   1.3499603 -0.4262852
## 2   2.3116706  -0.3415252 -0.04902063 -0.1411004  -0.7405087 -0.4262852
## 3  -0.4324394  -0.3415252 -0.04902063 -0.1411004   1.3499603 -0.4262852
## 4  -0.4324394  -0.3415252 -0.04902063 -0.1411004  -0.7405087  2.3450435
## 5  -0.4324394  -0.3415252 -0.04902063 -0.1411004   1.3499603 -0.4262852
```

```
## 6  -0.4324394  -0.3415252 -0.04902063 -0.1411004   1.3499603 -0.4262852
##   Ext_WdShing  Ext_BrkCmn Ext_CmentBd Ext_Other Ext_WdShng     BF_ALQ
## 1  -0.1398328 -0.08712899  -0.2123613 -0.018509 -0.1689125 -0.3898471
## 2  -0.1398328 -0.08712899  -0.2123613 -0.018509 -0.1689125  3.3760895
## 3  -0.1398328 -0.08712899  -0.2123613 -0.018509 -0.1689125 -0.3898471
## 4  -0.1398328 -0.08712899  -0.2123613 -0.018509  5.9181952  0.4418935
## 5  -0.1398328 -0.08712899  -0.2123613 -0.018509 -0.1689125 -0.3898471
## 6  -0.1398328 -0.08712899  -0.2123613 -0.018509 -0.1689125 -0.3898471
##       BF_BLQ     BF_GLQ     BF_LwQ     BF_Rec    BF_Unf Centroid_1 Centroid_2
## 1 -0.3089908  1.0182075 -0.2400835 -0.3259503 -0.018509 -0.4671947 -0.3487179
## 2 -0.3089908 -0.5358624 -0.2400835 -0.3259503 -0.018509 -0.5035475 -0.3813294
## 3 -0.3089908  0.5339364 -0.2400835 -0.3259503 -0.018509 -0.5942926 -0.4638718
## 4 -0.3089908 -0.5358624 -0.2400835 -0.3259503 -0.018509 -0.4786774 -0.3599285
## 5 -0.3089908  0.9059446 -0.2400835 -0.3259503 -0.018509 -1.7213829 -1.4901246
## 6 -0.3089908  1.0754395 -0.2400835 -0.3259503 -0.018509  0.6672481  0.6763999
##   Centroid_3 Centroid_4 Centroid_5 Centroid_6  Centroid_7 Centroid_8 Centroid_9
## 1 -0.1641580 -0.7179109  0.1600668 -0.2738175 -0.08918966 -1.0741668 -0.6332393
## 2 -0.3012205 -0.7612469  0.1859867 -0.3037083 -0.05932873 -1.1283645 -0.6036930
## 3 -0.5020915 -0.8641826  0.2643689 -0.3794210  0.03128165 -1.2221690 -0.4995336
## 4 -0.2903352 -0.7290439  0.1682936 -0.2854952 -0.07613038 -1.0691748 -0.6122911
## 5 -0.7398933 -1.4013686  1.2402524 -1.3137784  1.14521164 -0.1020894  0.7723570
## 6 -0.3688488  0.5661828 -0.7443270  0.6514820 -0.60962513  0.2881595 -0.1572243
##   Centroid_10 MSZoning_C(all) MSZoning_FV MSZoning_RH MSZoning_RL MSZoning_RM
## 1   0.2314623     -0.09292795  -0.2235685 -0.09478466   0.5372549  -0.4324394
## 2   0.2558390     -0.09292795  -0.2235685 -0.09478466   0.5372549  -0.4324394
## 3   0.3295552     -0.09292795  -0.2235685 -0.09478466   0.5372549  -0.4324394
## 4   0.2371232     -0.09292795  -0.2235685 -0.09478466   0.5372549  -0.4324394
## 5   1.2431437     -0.09292795  -0.2235685 -0.09478466   0.5372549  -0.4324394
## 6  -0.6573392     -0.09292795  -0.2235685 -0.09478466   0.5372549  -0.4324394
##   Street_Grvl Street_Pave Alley_Grvl Alley_Pave LotShape_IR1 LotShape_IR2
## 1 -0.06423825  0.06423825 -0.2070212 -0.1656675   -0.7042626   -0.1634722
## 2 -0.06423825  0.06423825 -0.2070212 -0.1656675   -0.7042626   -0.1634722
## 3 -0.06423825  0.06423825 -0.2070212 -0.1656675    1.4194384   -0.1634722
## 4 -0.06423825  0.06423825 -0.2070212 -0.1656675    1.4194384   -0.1634722
## 5 -0.06423825  0.06423825 -0.2070212 -0.1656675    1.4194384   -0.1634722
## 6 -0.06423825  0.06423825 -0.2070212 -0.1656675    1.4194384   -0.1634722
##   LotShape_IR3 LotShape_Reg Utilities_AllPub Utilities_NoSeWa LotConfig_Corner
## 1  -0.07422703    0.7549859       0.03206952        -0.018509       -0.4605829
## 2  -0.07422703    0.7549859       0.03206952        -0.018509       -0.4605829
## 3  -0.07422703   -1.3240743       0.03206952        -0.018509       -0.4605829
## 4  -0.07422703   -1.3240743       0.03206952        -0.018509        2.1704180
## 5  -0.07422703   -1.3240743       0.03206952        -0.018509       -0.4605829
## 6  -0.07422703   -1.3240743       0.03206952        -0.018509       -0.4605829
##   LotConfig_CulDSac LotConfig_FR2 LotConfig_FR3 LotConfig_Inside
## 1        -0.2532614     -0.173155   -0.06940912         0.606934
## 2        -0.2532614      5.773193   -0.06940912        -1.647061
## 3        -0.2532614     -0.173155   -0.06940912         0.606934
## 4        -0.2532614     -0.173155   -0.06940912        -1.647061
## 5        -0.2532614      5.773193   -0.06940912        -1.647061
## 6        -0.2532614     -0.173155   -0.06940912         0.606934
##   Neighborhood_Blmngtn Neighborhood_Blueste Neighborhood_BrDale
## 1          -0.09839671          -0.05862107          -0.1018855
## 2          -0.09839671          -0.05862107          -0.1018855
## 3          -0.09839671          -0.05862107          -0.1018855
## 4          -0.09839671          -0.05862107          -0.1018855
## 5          -0.09839671          -0.05862107          -0.1018855
## 6          -0.09839671          -0.05862107          -0.1018855
##   Neighborhood_BrkSide Neighborhood_ClearCr Neighborhood_CollgCr
## 1           -0.1959779           -0.1236896            3.1510604
## 2           -0.1959779           -0.1236896           -0.3172448
## 3           -0.1959779           -0.1236896            3.1510604
## 4           -0.1959779           -0.1236896           -0.3172448
## 5           -0.1959779           -0.1236896           -0.3172448
## 6           -0.1959779           -0.1236896           -0.3172448
##   Neighborhood_Crawfor Neighborhood_Edwards Neighborhood_Gilbert
```

```
## 1           -0.1912176          -0.2667738          -0.2447291
## 2           -0.1912176          -0.2667738          -0.2447291
## 3           -0.1912176          -0.2667738          -0.2447291
## 4            5.2278523          -0.2667738          -0.2447291
## 5           -0.1912176          -0.2667738          -0.2447291
## 6           -0.1912176          -0.2667738          -0.2447291
##   Neighborhood_IDOTRR Neighborhood_MeadowV Neighborhood_Mitchel
## 1          -0.1813765          -0.1132868          -0.2015634
## 2          -0.1813765          -0.1132868          -0.2015634
## 3          -0.1813765          -0.1132868          -0.2015634
## 4          -0.1813765          -0.1132868          -0.2015634
## 5          -0.1813765          -0.1132868          -0.2015634
## 6          -0.1813765          -0.1132868           4.9595195
##   Neighborhood_NAmes Neighborhood_NoRidge Neighborhood_NPkVill
## 1         -0.4229141          -0.1578646          -0.08910257
## 2         -0.4229141          -0.1578646          -0.08910257
## 3         -0.4229141          -0.1578646          -0.08910257
## 4         -0.4229141          -0.1578646          -0.08910257
## 5         -0.4229141           6.3323719          -0.08910257
## 6         -0.4229141          -0.1578646          -0.08910257
##   Neighborhood_NridgHt Neighborhood_NWAmes Neighborhood_OldTown
## 1           -0.2455142          -0.2167279          -0.2985775
## 2           -0.2455142          -0.2167279          -0.2985775
## 3           -0.2455142          -0.2167279          -0.2985775
## 4           -0.2455142          -0.2167279          -0.2985775
## 5           -0.2455142          -0.2167279          -0.2985775
## 6           -0.2455142          -0.2167279          -0.2985775
##   Neighborhood_Sawyer Neighborhood_SawyerW Neighborhood_Somerst
## 1          -0.2335237          -0.2114791          -0.2578243
## 2          -0.2335237          -0.2114791          -0.2578243
## 3          -0.2335237          -0.2114791          -0.2578243
## 4          -0.2335237          -0.2114791          -0.2578243
## 5          -0.2335237          -0.2114791          -0.2578243
## 6          -0.2335237          -0.2114791          -0.2578243
##   Neighborhood_StoneBr Neighborhood_SWISU Neighborhood_Timber
## 1           -0.1333279         -0.1292795          -0.1590004
## 2           -0.1333279         -0.1292795          -0.1590004
## 3           -0.1333279         -0.1292795          -0.1590004
## 4           -0.1333279         -0.1292795          -0.1590004
## 5           -0.1333279         -0.1292795          -0.1590004
## 6           -0.1333279         -0.1292795          -0.1590004
##   Neighborhood_Veenker HouseStyle_1.5Fin HouseStyle_1.5Unf HouseStyle_1Story
## 1          -0.09103469        -0.3471255       -0.08092886        -1.0077380
## 2          10.98105987        -0.3471255       -0.08092886         0.9919814
## 3          -0.09103469        -0.3471255       -0.08092886        -1.0077380
## 4          -0.09103469        -0.3471255       -0.08092886        -1.0077380
## 5          -0.09103469        -0.3471255       -0.08092886        -1.0077380
## 6          -0.09103469         2.8798153       -0.08092886        -1.0077380
##   HouseStyle_2.5Fin HouseStyle_2.5Unf HouseStyle_2Story HouseStyle_SFoyer
## 1       -0.05241426       -0.09103469         1.5318854        -0.1710455
## 2       -0.05241426       -0.09103469        -0.6525667        -0.1710455
## 3       -0.05241426       -0.09103469         1.5318854        -0.1710455
## 4       -0.05241426       -0.09103469         1.5318854        -0.1710455
## 5       -0.05241426       -0.09103469         1.5318854        -0.1710455
## 6       -0.05241426       -0.09103469        -0.6525667        -0.1710455
##   HouseStyle_SLvl MasVnrType_BrkCmn MasVnrType_BrkFace MasVnrType_None
## 1      -0.2141168       -0.09292795          1.5231625      -1.2163581
## 2      -0.2141168       -0.09292795         -0.6563038       0.8218447
## 3      -0.2141168       -0.09292795          1.5231625      -1.2163581
## 4      -0.2141168       -0.09292795         -0.6563038       0.8218447
## 5      -0.2141168       -0.09292795          1.5231625      -1.2163581
## 6      -0.2141168       -0.09292795         -0.6563038       0.8218447
##   MasVnrType_Stone Foundation_BrkTil Foundation_CBlock Foundation_PConc
## 1       -0.3053301        -0.3452646        -0.8562253        1.1096078
## 2       -0.3053301        -0.3452646         1.1675169       -0.9009106
## 3       -0.3053301        -0.3452646        -0.8562253        1.1096078
```

```
## 3       -0.3053301        -0.3452646        -0.8562253        1.1096078
## 4       -0.3053301         2.8953375        -0.8562253       -0.9009106
## 5       -0.3053301        -0.3452646        -0.8562253        1.1096078
## 6       -0.3053301        -0.3452646        -0.8562253       -0.9009106
##   Foundation_Slab Foundation_Stone Foundation_Wood BsmtExposure_Av
## 1       -0.130642      -0.06149287     -0.04141578      -0.4087492
## 2       -0.130642      -0.06149287     -0.04141578      -0.4087492
## 3       -0.130642      -0.06149287     -0.04141578      -0.4087492
## 4       -0.130642      -0.06149287     -0.04141578      -0.4087492
## 5       -0.130642      -0.06149287     -0.04141578       2.4456500
## 6       -0.130642      -0.06149287     24.13711546      -0.4087492
##   BsmtExposure_Gd BsmtExposure_Mn BsmtExposure_No Heating_Floor Heating_GasA
## 1       -0.323096      -0.2985775       0.7300038     -0.018509     0.125109
## 2        3.093995      -0.2985775      -1.3693865     -0.018509     0.125109
## 3       -0.323096       3.3480662      -1.3693865     -0.018509     0.125109
## 4       -0.323096      -0.2985775       0.7300038     -0.018509     0.125109
## 5       -0.323096      -0.2985775      -1.3693865     -0.018509     0.125109
## 6       -0.323096      -0.2985775       0.7300038     -0.018509     0.125109
##   Heating_GasW Heating_Grav Heating_OthW Heating_Wall Electrical_FuseA
## 1  -0.09660694  -0.05560327  -0.02618017   -0.0453765       -0.2623274
## 2  -0.09660694  -0.05560327  -0.02618017   -0.0453765       -0.2623274
## 3  -0.09660694  -0.05560327  -0.02618017   -0.0453765       -0.2623274
## 4  -0.09660694  -0.05560327  -0.02618017   -0.0453765       -0.2623274
## 5  -0.09660694  -0.05560327  -0.02618017   -0.0453765       -0.2623274
## 6  -0.09660694  -0.05560327  -0.02618017   -0.0453765       -0.2623274
##   Electrical_FuseF Electrical_FuseP Electrical_Mix Electrical_SBrkr
## 1       -0.1319913      -0.05241426      -0.018509        0.3046593
## 2       -0.1319913      -0.05241426      -0.018509        0.3046593
## 3       -0.1319913      -0.05241426      -0.018509        0.3046593
## 4       -0.1319913      -0.05241426      -0.018509        0.3046593
## 5       -0.1319913      -0.05241426      -0.018509        0.3046593
## 6       -0.1319913      -0.05241426      -0.018509        0.3046593
##   Functional_Maj1 Functional_Maj2 Functional_Min1 Functional_Min2
## 1     -0.08092886     -0.05560327      -0.1508882      -0.1567214
## 2     -0.08092886     -0.05560327      -0.1508882      -0.1567214
## 3     -0.08092886     -0.05560327      -0.1508882      -0.1567214
## 4     -0.08092886     -0.05560327      -0.1508882      -0.1567214
## 5     -0.08092886     -0.05560327      -0.1508882      -0.1567214
## 6     -0.08092886     -0.05560327      -0.1508882      -0.1567214
##   Functional_Mod Functional_Sev Functional_Typ GarageType_2Types
## 1     -0.1101443    -0.02618017      0.2726192       -0.08910257
## 2     -0.1101443    -0.02618017      0.2726192       -0.08910257
## 3     -0.1101443    -0.02618017      0.2726192       -0.08910257
## 4     -0.1101443    -0.02618017      0.2726192       -0.08910257
## 5     -0.1101443    -0.02618017      0.2726192       -0.08910257
## 6     -0.1101443    -0.02618017      0.2726192       -0.08910257
##   GarageType_Attchd GarageType_Basment GarageType_BuiltIn GarageType_CarPort
## 1         0.8330068         -0.1117261         -0.2608328        -0.07185764
## 2         0.8330068         -0.1117261         -0.2608328        -0.07185764
## 3         0.8330068         -0.1117261         -0.2608328        -0.07185764
## 4        -1.2000591         -0.1117261         -0.2608328        -0.07185764
## 5         0.8330068         -0.1117261         -0.2608328        -0.07185764
## 6         0.8330068         -0.1117261         -0.2608328        -0.07185764
##   GarageType_Detchd GarageFinish_Fin GarageFinish_RFn GarageFinish_Unf
## 1        -0.6032363       -0.5715822        1.6119459       -0.8532245
## 2        -0.6032363       -0.5715822        1.6119459       -0.8532245
## 3        -0.6032363       -0.5715822        1.6119459       -0.8532245
## 4         1.6571574       -0.5715822       -0.6201557        1.1716229
## 5        -0.6032363       -0.5715822        1.6119459       -0.8532245
## 6        -0.6032363       -0.5715822       -0.6201557        1.1716229
##   PavedDrive_N PavedDrive_P PavedDrive_Y MiscFeature_Gar2 MiscFeature_Othr
## 1   -0.2826373   -0.1472876    0.3243873      -0.04141578      -0.03703704
## 2   -0.2826373   -0.1472876    0.3243873      -0.04141578      -0.03703704
## 3   -0.2826373   -0.1472876    0.3243873      -0.04141578      -0.03703704
## 4   -0.2826373   -0.1472876    0.3243873      -0.04141578      -0.03703704
## 5   -0.2826373   -0.1472876    0.3243873      -0.04141578      -0.03703704
```

```
## 6   -0.2826373   -0.1472876    0.3243873      -0.04141578      -0.03703704
##   MiscFeature_Shed MiscFeature_TenC SaleType_COD SaleType_Con SaleType_ConLD
## 1       -0.1833813        -0.018509   -0.1752422  -0.04141578    -0.09478466
## 2       -0.1833813        -0.018509   -0.1752422  -0.04141578    -0.09478466
## 3       -0.1833813        -0.018509   -0.1752422  -0.04141578    -0.09478466
## 4       -0.1833813        -0.018509   -0.1752422  -0.04141578    -0.09478466
## 5       -0.1833813        -0.018509   -0.1752422  -0.04141578    -0.09478466
## 6        5.4512505        -0.018509   -0.1752422  -0.04141578    -0.09478466
##   SaleType_ConLI SaleType_ConLw SaleType_CWD SaleType_New SaleType_Oth
## 1    -0.05560327    -0.05241426  -0.06423825   -0.2985775  -0.05241426
## 2    -0.05560327    -0.05241426  -0.06423825   -0.2985775  -0.05241426
## 3    -0.05560327    -0.05241426  -0.06423825   -0.2985775  -0.05241426
## 4    -0.05560327    -0.05241426  -0.06423825   -0.2985775  -0.05241426
## 5    -0.05560327    -0.05241426  -0.06423825   -0.2985775  -0.05241426
## 6    -0.05560327    -0.05241426  -0.06423825   -0.2985775  -0.05241426
##   SaleType_WD SaleCondition_Abnorml SaleCondition_AdjLand SaleCondition_Alloca
## 1   0.3949508            -0.2638157           -0.06423825          -0.09103469
## 2   0.3949508            -0.2638157           -0.06423825          -0.09103469
## 3   0.3949508            -0.2638157           -0.06423825          -0.09103469
## 4   0.3949508             3.7892265           -0.06423825          -0.09103469
## 5   0.3949508            -0.2638157           -0.06423825          -0.09103469
## 6   0.3949508            -0.2638157           -0.06423825          -0.09103469
##   SaleCondition_Family SaleCondition_Normal SaleCondition_Partial Id
## 1           -0.1265135            0.4638573            -0.3026411  1
## 2           -0.1265135            0.4638573            -0.3026411  2
## 3           -0.1265135            0.4638573            -0.3026411  3
## 4           -0.1265135           -2.1550970            -0.3026411  4
## 5           -0.1265135            0.4638573            -0.3026411  5
## 6           -0.1265135            0.4638573            -0.3026411  6
##   SalePrice_Log
## 1      12.24769
## 2      12.10901
## 3      12.31717
## 4      11.84940
## 5      12.42922
## 6      11.87060
```

## Model learning

Once we are done with data wrangling, we gan step forward to model training.

### Boosted Generalized Linear Model

```
glm_boost_every_model
```

```
## Boosted Generalized Linear Model
##
## 1460 samples
##  208 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1314, 1313, 1315, 1314, 1312, 1315, ...
## Resampling results across tuning parameters:
##
##   mstop  RMSE       Rsquared   MAE
##    50    0.1549680  0.8563427  0.10296030
##   100    0.1463509  0.8678162  0.09685971
##   150    0.1436602  0.8721018  0.09441249
##
## Tuning parameter 'prune' was held constant at a value of no
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were mstop = 150 and prune = no.
```

```
varImp(glm_boost_every_model)
```

```
## glmboost variable importance
##
##   only 20 most important variables shown (out of 209)
##
##                          Overall
## OverallQual             100.000
## TotalSquare              62.888
## Centroid_2               36.810
## Bathrooms                31.455
## GarageCars               30.048
## LotArea_log              28.943
## Freshness                20.852
## `\\`MSZoning_C(all)\\``  16.369
## KitchenQual              12.178
## MSZoning_RM              11.877
## CentralAir               11.738
## FireplaceQu              10.669
## Neighborhood_Crawfor      9.233
## Age                       9.197
## HeatingQC                 8.881
## SaleCondition_Abnorml     7.799
## OverallCond               7.639
## PorchArea                 5.639
## Functional_Maj2           5.434
## Neighborhood_Edwards      5.187
```

### Gaussian Process with Polynomial Kernel

```
gauss_process_poly_model
```

```
## Gaussian Process with Polynomial Kernel
##
## 1460 samples
##  208 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1314, 1313, 1315, 1314, 1312, 1315, ...
## Resampling results across tuning parameters:
##
##   degree  scale  RMSE        Rsquared     MAE
##   1       0.001   0.1342657  0.888433001  0.08735820
##   1       0.010   0.1375496  0.883063111  0.08855159
##   1       0.100   0.1678613  0.834687391  0.09734484
##   2       0.001   0.1551244  0.853003518  0.09599557
##   2       0.010   0.1858059  0.808879959  0.12297639
##   2       0.100   1.2992393  0.124343806  0.80344763
##   3       0.001   0.3186871  0.580759788  0.19624857
##   3       0.010  15.7502751  0.006718686  8.66911418
##   3       0.100         NaN          NaN         NaN
##
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were degree = 1 and scale = 0.001.
```

```
varImp(gauss_process_poly_model)
```

```
## loess r-squared variable importance

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

##   only 20 most important variables shown (out of 208)
##
##             Overall
## TotalSquare  100.00
## OverallQual   97.81
## OverallWow    96.46
## Centroid_10   93.41
## Centroid_6    93.15
## Centroid_2    87.41
## Centroid_1    80.15
## Centroid_5    79.51
## GrLivArea     79.00
## GarageCars    67.85
## ExterQual     67.50
## GarageWow     66.66
## Bathrooms     66.34
## GarageArea    65.80
## KitchenQual   65.34
## TotalBsmtSF   62.26
## Centroid_4    56.58
## Freshness     56.14
## BsmtQual      55.54
## Overall       53.94
```

## Random Forest

```
forest_model
```

```
## Random Forest
##
## 1460 samples
##  208 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1314, 1313, 1315, 1314, 1312, 1315, ...
## Resampling results across tuning parameters:
##
##   mtry  RMSE       Rsquared   MAE
##     2   0.1731576  0.8756816  0.11709864
##   105   0.1335062  0.8905046  0.08827383
##   208   0.1362422  0.8850440  0.09064672
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was mtry = 105.
```

```
varImp(forest_model)
```

```
## rf variable importance
##
##   only 20 most important variables shown (out of 208)
##
##             Overall
## OverallQual 100.000
## TotalSquare  79.575
## Centroid_10  30.445
## Centroid_5   20.484
## OverallWow   17.193
## Centroid_6   11.886
## Age           9.454
## Centroid_3    8.778
## TotalBsmtSF   6.594
## Centroid_2    6.491
## Centroid_1    5.540
## GarageCars    5.120
## BasementWow   4.921
## Centroid_9    4.465
## GarageWow     3.816
## Freshness     3.747
## GrLivArea     3.417
## Centroid_8    3.087
## Centroid_4    3.072
## Centroid_7    3.056
```

### eXtreme Gradient Boosting

```
tree_model
```

```
## eXtreme Gradient Boosting
##
## 1460 samples
##  208 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1314, 1313, 1315, 1314, 1312, 1315, ...
## Resampling results across tuning parameters:
##
##   eta  max_depth  colsample_bytree  subsample  nrounds  RMSE       Rsquared
##   0.3  1          0.6               0.50        50      0.1464181  0.8679102
##   0.3  1          0.6               0.50       100      0.1371464  0.8835716
##   0.3  1          0.6               0.50       150      0.1340310  0.8891321
##   0.3  1          0.6               0.75        50      0.1499860  0.8609337
##   0.3  1          0.6               0.75       100      0.1392651  0.8801095
##   0.3  1          0.6               0.75       150      0.1359688  0.8858246
##   0.3  1          0.6               1.00        50      0.1467883  0.8665469
##   0.3  1          0.6               1.00       100      0.1369485  0.8839422
##   0.3  1          0.6               1.00       150      0.1334774  0.8896122
##   0.3  1          0.8               0.50        50      0.1485460  0.8642289
##   0.3  1          0.8               0.50       100      0.1383752  0.8817248
##   0.3  1          0.8               0.50       150      0.1345998  0.8878806
##   0.3  1          0.8               0.75        50      0.1489561  0.8627248
##   0.3  1          0.8               0.75       100      0.1387091  0.8812733
##   0.3  1          0.8               0.75       150      0.1341980  0.8886350
##   0.3  1          0.8               1.00        50      0.1483244  0.8639706
##   0.3  1          0.8               1.00       100      0.1376474  0.8827596
##   0.3  1          0.8               1.00       150      0.1335126  0.8894170
##   0.3  2          0.6               0.50        50      0.1385624  0.8816737
##   0.3  2          0.6               0.50       100      0.1341442  0.8888962
##   0.3  2          0.6               0.50       150      0.1347937  0.8884740
##   0.3  2          0.6               0.75        50      0.1343457  0.8871972
```

```
##   0.3  2          0.6               0.75      100      0.1297567  0.8952664
##   0.3  2          0.6               0.75      150      0.1282172  0.8977153
##   0.3  2          0.6               1.00       50      0.1342800  0.8890765
##   0.3  2          0.6               1.00      100      0.1297713  0.8965579
##   0.3  2          0.6               1.00      150      0.1289426  0.8976606
##   0.3  2          0.8               0.50       50      0.1399418  0.8790214
##   0.3  2          0.8               0.50      100      0.1366377  0.8848303
##   0.3  2          0.8               0.50      150      0.1361929  0.8860201
##   0.3  2          0.8               0.75       50      0.1319469  0.8931042
##   0.3  2          0.8               0.75      100      0.1277146  0.8996108
##   0.3  2          0.8               0.75      150      0.1276327  0.8995881
##   0.3  2          0.8               1.00       50      0.1345396  0.8879146
##   0.3  2          0.8               1.00      100      0.1307061  0.8941392
##   0.3  2          0.8               1.00      150      0.1303094  0.8947635
##   0.3  3          0.6               0.50       50      0.1354822  0.8854342
##   0.3  3          0.6               0.50      100      0.1349622  0.8872944
##   0.3  3          0.6               0.50      150      0.1349464  0.8871158
##   0.3  3          0.6               0.75       50      0.1380142  0.8828150
##   0.3  3          0.6               0.75      100      0.1364893  0.8856411
##   0.3  3          0.6               0.75      150      0.1368324  0.8854927
##   0.3  3          0.6               1.00       50      0.1315929  0.8926247
##   0.3  3          0.6               1.00      100      0.1300470  0.8950177
##   0.3  3          0.6               1.00      150      0.1306209  0.8940684
##   0.3  3          0.8               0.50       50      0.1359763  0.8841136
##   0.3  3          0.8               0.50      100      0.1359592  0.8849682
##   0.3  3          0.8               0.50      150      0.1353584  0.8861132
##   0.3  3          0.8               0.75       50      0.1355338  0.8868962
##   0.3  3          0.8               0.75      100      0.1345935  0.8886663
##   0.3  3          0.8               0.75      150      0.1346468  0.8885893
##   0.3  3          0.8               1.00       50      0.1311656  0.8937873
##   0.3  3          0.8               1.00      100      0.1297991  0.8960402
##   0.3  3          0.8               1.00      150      0.1299899  0.8959507
##   0.4  1          0.6               0.50       50      0.1558848  0.8526675
##   0.4  1          0.6               0.50      100      0.1430289  0.8757609
##   0.4  1          0.6               0.50      150      0.1406146  0.8791963
##   0.4  1          0.6               0.75       50      0.1513155  0.8583963
##   0.4  1          0.6               0.75      100      0.1415615  0.8761908
##   0.4  1          0.6               0.75      150      0.1366730  0.8844329
##   0.4  1          0.6               1.00       50      0.1488521  0.8633794
##   0.4  1          0.6               1.00      100      0.1373292  0.8834816
##   0.4  1          0.6               1.00      150      0.1325117  0.8915161
##   0.4  1          0.8               0.50       50      0.1469896  0.8651698
##   0.4  1          0.8               0.50      100      0.1387410  0.8802157
##   0.4  1          0.8               0.50      150      0.1331353  0.8894248
##   0.4  1          0.8               0.75       50      0.1498279  0.8614664
##   0.4  1          0.8               0.75      100      0.1378995  0.8824084
##   0.4  1          0.8               0.75      150      0.1347467  0.8878063
##   0.4  1          0.8               1.00       50      0.1501111  0.8594539
##   0.4  1          0.8               1.00      100      0.1376773  0.8818849
##   0.4  1          0.8               1.00      150      0.1323384  0.8907047
##   0.4  2          0.6               0.50       50      0.1435249  0.8741407
##   0.4  2          0.6               0.50      100      0.1405058  0.8793366
##   0.4  2          0.6               0.50      150      0.1399298  0.8810129
##   0.4  2          0.6               0.75       50      0.1369230  0.8845503
##   0.4  2          0.6               0.75      100      0.1359154  0.8860755
##   0.4  2          0.6               0.75      150      0.1353810  0.8876796
##   0.4  2          0.6               1.00       50      0.1344032  0.8888063
##   0.4  2          0.6               1.00      100      0.1308896  0.8945656
##   0.4  2          0.6               1.00      150      0.1297126  0.8965912
##   0.4  2          0.8               0.50       50      0.1451708  0.8712396
##   0.4  2          0.8               0.50      100      0.1423365  0.8763585
##   0.4  2          0.8               0.50      150      0.1428546  0.8752525
##   0.4  2          0.8               0.75       50      0.1352037  0.8874546
##   0.4  2          0.8               0.75      100      0.1349646  0.8882672
##   0.4  2          0.8               0.75      150      0.1360046  0.8867182
##   0.4  2          0.8               1.00       50      0.1354773  0.8862467
```

```
##   0.4  2          0.8                1.00       100      0.1312183  0.8929412
##   0.4  2          0.8                1.00       150      0.1305450  0.8941875
##   0.4  3          0.6                0.50        50      0.1439642  0.8728094
##   0.4  3          0.6                0.50       100      0.1463541  0.8697985
##   0.4  3          0.6                0.50       150      0.1462369  0.8697127
##   0.4  3          0.6                0.75        50      0.1443540  0.8734184
##   0.4  3          0.6                0.75       100      0.1442883  0.8737887
##   0.4  3          0.6                0.75       150      0.1439223  0.8746597
##   0.4  3          0.6                1.00        50      0.1357226  0.8867304
##   0.4  3          0.6                1.00       100      0.1359903  0.8864323
##   0.4  3          0.6                1.00       150      0.1374004  0.8842237
##   0.4  3          0.8                0.50        50      0.1477549  0.8683875
##   0.4  3          0.8                0.50       100      0.1466933  0.8703731
##   0.4  3          0.8                0.50       150      0.1482988  0.8675780
##   0.4  3          0.8                0.75        50      0.1343174  0.8882168
##   0.4  3          0.8                0.75       100      0.1343720  0.8879586
##   0.4  3          0.8                0.75       150      0.1343698  0.8880637
##   0.4  3          0.8                1.00        50      0.1348906  0.8879447
##   0.4  3          0.8                1.00       100      0.1332277  0.8904566
##   0.4  3          0.8                1.00       150      0.1327212  0.8911380
##   MAE
##   0.10416732
##   0.09563717
##   0.09345769
##   0.10783148
##   0.09776157
##   0.09310532
##   0.10547665
##   0.09623541
##   0.09269459
##   0.10742580
##   0.09801693
##   0.09409535
##   0.10697989
##   0.09759163
##   0.09285784
##   0.10610034
##   0.09657963
##   0.09246369
##   0.09760571
##   0.09184273
##   0.09109984
##   0.09379710
##   0.08868812
##   0.08786008
##   0.09326566
##   0.08805759
##   0.08721383
##   0.09756889
##   0.09299093
##   0.09201271
##   0.09278796
##   0.08816667
##   0.08757170
##   0.09326259
##   0.08806307
##   0.08659726
##   0.09551203
##   0.09439155
##   0.09488613
##   0.09444676
##   0.09212098
##   0.09226764
##   0.09078466
##   0.08916215
```

```
##   0.09001065
##   0.09367873
##   0.09330117
##   0.09193600
##   0.09231276
##   0.09061759
##   0.09038853
##   0.09043232
##   0.08846093
##   0.08890262
##   0.11071098
##   0.10123819
##   0.09776711
##   0.10983603
##   0.09945585
##   0.09498801
##   0.10879670
##   0.09831847
##   0.09323041
##   0.10764569
##   0.09892250
##   0.09444354
##   0.10610610
##   0.09584531
##   0.09189781
##   0.10869205
##   0.09878102
##   0.09400881
##   0.09785495
##   0.09558947
##   0.09508364
##   0.09435657
##   0.09190738
##   0.09100157
##   0.09412664
##   0.08989165
##   0.08871078
##   0.09999862
##   0.09802093
##   0.09766368
##   0.09473116
##   0.09248884
##   0.09208133
##   0.09320469
##   0.08932179
##   0.08850499
##   0.10101014
##   0.10243042
##   0.10251130
##   0.09775181
##   0.09796856
##   0.09812833
##   0.09194703
##   0.09149043
##   0.09276295
##   0.10125059
##   0.10059827
##   0.10258869
##   0.09427051
##   0.09426125
##   0.09506310
##   0.09407775
##   0.09247454
##   0.09222468
##
## Tuning parameter 'gamma' was held constant at a value of 0
```

```
## Tuning parameter 'gamma' was held constant at a value of 0
## Tuning
##  parameter 'min_child_weight' was held constant at a value of 1
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were nrounds = 150, max_depth = 2, eta
##  = 0.3, gamma = 0, colsample_bytree = 0.8, min_child_weight = 1 and subsample
##  = 0.75.
```

```
varImp(tree_model)
```

```
## xgbTree variable importance
##
##   only 20 most important variables shown (out of 208)
##
##                   Overall
## OverallQual       100.000
## Centroid_10        87.455
## TotalSquare        71.354
## Age                66.987
## GarageWow          51.968
## Bathrooms          23.002
## Centroid_7         14.752
## GrLivArea          14.703
## Freshness          13.779
## BasementWow        12.965
## LotArea            10.954
## Overall            10.822
## GarageType_Attchd  10.491
## `MSZoning_C(all)`   8.806
## Centroid_9          7.636
## PorchArea           7.407
## Centroid_6          6.896
## GarageCars          6.353
## GarageYrBlt         4.208
## TotalBsmtSF         3.933
```

### Bayesian Regularized Neural Networks

For better fit, we will need to choose most important variables for learning neural networks.

```
bayes_neural_model
```

```
## Bayesian Regularized Neural Networks
##
## 1460 samples
##   27 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1314, 1313, 1315, 1314, 1312, 1315, ...
## Resampling results across tuning parameters:
##
##   neurons  RMSE       Rsquared   MAE
##   1        0.1474489  0.8652179  0.09897360
##   2        0.1415625  0.8762793  0.09538935
##   3        0.1366794  0.8836342  0.09455310
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was neurons = 3.
```

### Elasticnet

```r
enet_model
```

```
## Elasticnet
##
## 1460 samples
##  208 predictor
##
## No pre-processing
## Resampling: Cross-Validated (10 fold)
## Summary of sample sizes: 1314, 1313, 1315, 1314, 1312, 1315, ...
## Resampling results across tuning parameters:
##
##   lambda  fraction  RMSE          Rsquared   MAE
##   0e+00   0.050     2.295155e+27  0.5362956  1.969337e+26
##   0e+00   0.525     2.409913e+28  0.5366722  2.067804e+27
##   0e+00   1.000     4.590310e+28  0.5045070  3.938674e+27
##   1e-04   0.050     2.667544e-01  0.7616869  2.000348e-01
##   1e-04   0.525     1.525656e-01  0.8541499  9.014215e-02
##   1e-04   1.000     3.442725e+01  0.4287008  3.919634e+00
##   1e-01   0.050     3.443326e-01  0.7190620  2.641439e-01
##   1e-01   0.525     1.396998e-01  0.8817236  9.011343e-02
##   1e-01   1.000     1.806457e+00  0.4468915  2.513232e-01
##
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were fraction = 0.525 and lambda = 0.1.
```

```r
varImp(enet_model)
```

```
## loess r-squared variable importance

## Warning in FUN(newX[, i], ...): no non-missing arguments to max; returning -Inf

##   only 20 most important variables shown (out of 208)
##
##             Overall
## TotalSquare  100.00
## OverallQual   97.81
## OverallWow    96.46
## Centroid_10   93.41
## Centroid_6    93.15
## Centroid_2    87.41
## Centroid_1    80.15
## Centroid_5    79.51
## GrLivArea     79.00
## GarageCars    67.85
## ExterQual     67.50
## GarageWow     66.66
## Bathrooms     66.34
## GarageArea    65.80
## KitchenQual   65.34
## TotalBsmtSF   62.26
## Centroid_4    56.58
## Freshness     56.14
## BsmtQual      55.54
## Overall       53.94
```

## Results

Firstly, I will run the models on the training set and plot results.

```
glm_boost_every_result_test <- predict(glm_boost_every_model, engineered_train_set, type = "raw")
gauss_process_poly_result_test <- predict(gauss_process_poly_model, engineered_train_set, type = "raw")
rf_result_test <- predict(forest_model, engineered_train_set, type = "raw")
boost_tree_result_test <- predict(tree_model, engineered_train_set, type = "raw")
bayes_neural_result_test <- predict(bayes_neural_model, engineered_train_set, type = "raw")
enet_result_test <- predict(enet_model, engineered_train_set, type = "raw")

voting_result_test <- (glm_boost_every_result_test + gauss_process_poly_result_test  + rf_result_test  + boost_tree_result_test + ba
test_rmse <- c(
  RMSE(engineered_train_set$SalePrice_Log,glm_boost_every_result_test),
  RMSE(engineered_train_set$SalePrice_Log,gauss_process_poly_result_test),
  RMSE(engineered_train_set$SalePrice_Log,rf_result_test),
  RMSE(engineered_train_set$SalePrice_Log,boost_tree_result_test),
  RMSE(engineered_train_set$SalePrice_Log,bayes_neural_result_test),
  RMSE(engineered_train_set$SalePrice_Log,enet_result_test),
  RMSE(engineered_train_set$SalePrice_Log,voting_result_test)
  )

data.frame(model = c("glm_boost", "gauss_poly","rf", "boost_tree","bayes_nn", "elasticnet" , "voting"), test_rmse = test_rmse) %>%
  ggplot(aes(x = model, y = test_rmse, label = model)) +
  geom_point() +
    geom_text(hjust=0, vjust=0)
```

In order to validate the resulting models, the estimations should be uploaded to kaggle.com, so the RMSE will be written manually by myself. The real result can be checked on kaggle.com leaderboard(my nickname is bombila78)

```
engineered_goal_set <- engineered_whole_set %>% filter(SalePrice_Log == 0)

glm_boost_every_result <- predict(glm_boost_every_model, engineered_goal_set, type = "raw")
gauss_process_poly_result <- predict(gauss_process_poly_model, engineered_goal_set, type = "raw")
rf_result <- predict(forest_model, engineered_goal_set, type = "raw")
boost_tree_result <- predict(tree_model, engineered_goal_set, type = "raw")
bayes_neural_result <- predict(bayes_neural_model, engineered_goal_set, type = "raw")
enet_result <- predict(enet_model, engineered_goal_set, type = "raw")
voting_result <- (glm_boost_every_result + gauss_process_poly_result  + rf_result  + boost_tree_result + bayes_neural_result + enet_

write.csv(data.frame(id=engineered_goal_set$Id, SalePrice=exp(glm_boost_every_result)), "estimations/glm_boost.csv", row.names = F)
write.csv(data.frame(id=engineered_goal_set$Id, SalePrice=exp(gauss_process_poly_result)), "estimations/gauss_poly.csv", row.names =
write.csv(data.frame(id=engineered_goal_set$Id, SalePrice=exp(rf_result)), "estimations/rf.csv", row.names = F)
write.csv(data.frame(id=engineered_goal_set$Id, SalePrice=exp(boost_tree_result)), "estimations/boost_tree.csv", row.names = F)
write.csv(data.frame(id=engineered_goal_set$Id, SalePrice=exp(bayes_neural_result)), "estimations/bayess_nn.csv", row.names = F)
write.csv(data.frame(id=engineered_goal_set$Id, SalePrice=exp(enet_result)), "estimations/enet.csv", row.names = F)
write.csv(data.frame(id=engineered_goal_set$Id, SalePrice=exp(voting_result)), "estimations/voting.csv", row.names = F)


validation_rmse <- c(0.13603, 0.13501, 0.12876, 0.12872, 0.13877, 0.13350, 0.12212)

data.frame(model = c("glm_boost", "gauss_poly","rf", "boost_tree","bayes_nn", "elasticnet" , "voting"), validation_rmse = validation
  ggplot(aes(x = model, y = validation_rmse, label = model)) +
  geom_point() +
  geom_text(hjust=0, vjust=0)
```

The resulting score is **0.12212**, that is pretty good and allows to be in top-500 out of 5000+ competitors.